УДК 004.89:004.912

ИСПОЛЬЗОВАНИЕ МЕТОДОВ ВЕКТОРИЗАЦИИ ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ ДЛЯ ПОВЫШЕНИЯ КАЧЕСТВА КОНТЕНТНЫХ РЕКОМЕНДАЦИЙ ФИЛЬМОВ

Федоренко В.И., Киреев В.С.

ФГАОУ Национальный исследовательский ядерный университет «МИФИ», Москва, e-mail: vladfedorenko94@gmail.com, vskireev@mephi.ru

Рекомендательные системы становятся незаменимыми компонентами любой веб-системы, предлагающей пользователям контент. Одной из актуальных задач в области построения контентной фильтрации является задача автоматического формирования признакового описания объектов системы. Для составления признаков текста, например аннотации фильма, требуется специальный метод предобработки – векторизация. В данной работе приводится сравнение методов построения векторных репрезентаций текстов на естественном языке для повышения качества рекомендаций фильмов. Описываются современные методы векторизации, такие как «мешок слов», латентно-семантическое индексирование, основанное на сингулярной декомпозиции. Так же предложен подход построения персональных рекомендаций на основе модели дистрибутивной семантики word2vec. Предложенный подход оценивается с помощью известных метрик качества машинного обучения – F-меры и других, а также перекрёстной проверки на 5 блоках. В качестве данных для экспериментов был выбран набор MovieLens, содержащий 1 млн пользовательских оценок. Для составления текстового корпуса использовались данные IMDB, по более чем 500 тыс. фильмов. Приведены полученные с помощью предложенного подхода семантически близкие слова и выработанные таким образом ключевые слова для фильмов разных жанров.

Ключевые слова: word2vec, векторные представления текстов, рекомендательные системы, контентная фильтрация, анализ текстов на естественном языке

TEXT EMBEDDINGS FOR CONTENT-BASED RECOMMENDATIONS

Fedorenko V.I., Kireev V.S.

National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow, e-mail: vladfedorenko94@gmail.com, vskireev@mephi.ru

Recommender systems are becoming indispensable components of any web-based system, offering users of the content. One of the actual tasks in the field of content filtering is the problem of automatic formation of a feature description of the system objects. To compile features of the text, for example, the annotation of the film requires a special method of pre-processing – vectorization. This paper presents a comparison of methods for constructing vector representations of natural language texts to improve the quality of film recommendations. Describes modern methods of vectorization, such as «bag of words» latent semantic indexing based on singular value decomposition. Also, the approach of building personal recommendations based on the model of word2vec distributive semantics is proposed. The proposed approach is evaluated using the well-known quality metrics machine learning F-measure, as well as cross-validation for 5 blocks. A set of MovieLens containing 1 million user ratings was chosen as the data for experiments. IMDB data for more than 500 thousand films were used to compile the text body. The author presents the semantically close words obtained with the help of the proposed approach and the keywords developed in this way for films of different genres.

Keywords: word2vec, text embeddings, recommender systems, content-based filtering, natural language processing

В настоящее время существенно растут объёмы информации и количество услуг, доступных пользователям. Количество услуг настолько велико, что пользователь физически не способен рассмотреть все доступные предложения. Поэтому рекомендательные системы становятся одним из незаменимых компонентов веб-приложений, предлагающих пользователю услуги.

Одной из актуальных задач в области рекомендательных систем является выделение признаковых описаний объектов. В данной работе рассмотрены методы построения векторных представлений текстов на естественном языке для увеличения качества контентной рекомендательной системы для фильмов на основе их сюжетных аннотаций, а также предложен подход построения контентных рекомендаций на основе модели word2vec.

Задача построения персональных рекомендаций

Франческо Ричи определил рекомендательные системы как методы и инструменты для предложения пользователям объектов для их использования. Формально задача ставится следующим образом: имеется множество пользователей $u_i \in U\,(i=1..n)$, множество объектов $i_j \in I\,(j=1..m)$ и множество оценок $r_{ij} \in R$. Необходимо для пользователя U' предложить набор объектов $\{i_j\}' \subseteq I$ так, чтобы максимизировать функ-

ционал качества Q. Бизнес-смысл данной метрики зависит от конкретной постановки задачи: данная метрика может отвечать как за удовлетворённость пользователя, так и за максимизацию прибыли владельца услуг.

Контентная фильтрация. Главная идея метода контентной фильтрации заключается в том, что пользователям интересны объекты, похожие на те, что им уже были интересны. В рамках подхода предполагается наличие полуформализованного признакового описания объектов рекомендаций, на основе которых возможно их взаимное сравнение и установление «похожести», например в случае книг, характеризовать и сравнивать объекты можно на основе аннотаций, жанра, автора, тематики, года издания и т.п. История пользовательской активности формирует профиль, в котором содержится информация о его предпочтениях [1]. Для определения сходства между векторами объектов чаще всего используется косинусное расстояние, принимающие значения в отрезке [0, 1]:

$$\cos(x,y)=\frac{A|B}{A\ B}=\frac{\sum_i A_i B_i}{\sqrt{\sum_i A_i^2}\sqrt{\sum_i B_i^2}}. \quad (1)$$

Если компоненты вектора не являются вещественными числами, то для оценки сходства можно выбрать расстояние Хемминга, определяемое как количество компонент, в которых различаются значения векторов

$$H(x,y)=\sum_{i=1}^{n}[x_i <> y_i]. \quad (2)$$

Метрики качества персональных рекомендаций. Наиболее распространенными метриками оценки качества рекомендательных систем являются точность и полнота [2]. Точность (*precision*) определяется как доля интересных пользователю объектов среди предложенных рекомендаций. Полнота (*recall*) определяется, как доля рекомендованных объектов среди всех интересных пользователю. В теории можно рекомендовать пользователю все объекты системы и получить идеальную полноту, на практике размер рекомендательного блока ограничен и поэтому для оценки фиксируют число k – размер блока рекомендаций и рассматривают точность и полноту $ap@k$ на k отобранных объектах. Итоговое качество модели определяется как среднее $ap@k$ по всем объектам – *MeanAveragePrecision@k.*

В целях учёта обоих характеристик для оценки также используют F_1-меру, равную среднему гармоническому полноты и точности:

$$F_1=2\frac{precision\cdot recall}{precision+recall}. \quad (3)$$

Альтернативным способом оценки модели является оценка качества восстановления рейтингов, которыми пользователь оценил объекты. В данном случае чаще всего рассматривается среднеквадратичное отклонение RMSE:

$$RMSE=\sqrt{\frac{\sum_{(i,j)\in R}\left(\hat{r}_{ij}-r_{ij}\right)^2}{n}}, \quad (4)$$

где n – число оценок.

Способы представления текстов на естественном языке

Основными сущностями при построении векторных представлений текстов на естественном языке являются токен t, документ d, корпус D и словарь V. Токеном называется элементарная единица текста, как правило, токены являются словами. Документом называется упорядоченный набор токенов, документом может являться статья, предложение, отзыв, описание услуги. Корпусом называется совокупность всех доступных документов. Словарём называют множество всех уникальных токенов, встречающихся в корпусе. Каждому токену ставится в соответствие уникальный индекс от 1 до $|V|$, таким образом, после процедуры построения словаря доступны два отображения: индекс -> токен, токен -> индекс

Для уменьшения размера словаря, сокращения количества вычислений и улучшения качества векторных представлений текстов на естественном языке используют методы предобработки текстов, такие как нормализация токенов, использование n-грамм и удаление слишком редких и слишком частых слов.

Стеммингом называется эвристическая процедура отбрасывания окончания слова и суффиксов. По сути, стемминг является определением неизменяющейся части слова. Альтернативным, более точным подходом является лемматизация, включающая в себя использование лексикографических словарей и морфологического анализа. Лемматизация является более вычислительно дорогостоящей операцией, чем стемминг [3].

Существуют слова, которые часто встречаются в тексте, но не несут серьезной смысловой нагрузки, к таким словам относятся предлоги, союзы, слова-обороты. В области анализа текстов на естественном языке их принято называть стоп-словами. Другой крайностью являются слишком редкие слова, которые встречаются недостаточ-

но часто для корректной оценки значимости токена в документе.

Помимо анализа токенов, часто рассматриваются *n*-граммы – последовательности из *n* подряд идущих токенов. Наиболее часто на практике используются биграммы и триграммы.

Модель мешка слов. Одним из наиболее простых методов представлений текстов на естественном языке является подход «мешок слов». Название «мешок» происходит из-за игнорирования порядка токенов в рассматриваемом документе. Так, два документа, отличающиеся лишь порядком токенов, будут иметь одинаковые векторы. Считается, что человек способен определить тематику текста, даже если слова будут перемешаны случайным образом. Каждый документ представляется вектором размерности $|V|$, где *i*-ая компонента вектора является счетчиком встречаемости *i*-ого токена в рассматриваемом документе, если какой-либо токен не встретился в рассматриваемом документе, то соответствующая компонента будет равна нулю.

TF-IDF. TF-IDF (Term Frequency. – Inverse Document Frequency) является модификацией мешка слов, в основу которой положено предположение, что токены в документе имеют разную значимость, и если слово встречается в небольшом числе документов, то оно является важным для них. TF вычисляется, как доля документов, в которых присутствует токен, а IDF как инверсия частоты, с которой некоторое слово встречается в документах коллекции. Вес токена в документе вычисляется как произведение TF*IDF.

$$tf(t,d) = \frac{n_t}{\sum_k n_k}, \tag{5}$$

$$idf(t,D) = log\frac{|D|}{\left|\{d_i \in D | t_i \in d_i\}\right|}, \tag{6}$$

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,D). \tag{7}$$

Тематическое моделирование

Тематическое моделирование является способом построения модели коллекции текстовых документов, которая определяет, к каким темам относится каждый из документов. Одной из первых моделей, предложенных в области тематического моделирования, является метод Latent Semantic Indexing (LSI), который заключается в усечённом SVD-разложении матрицы мешка слов до матрицы ранга d ($d << |D|$). После данного преобразования вектор документа будет состоять из d компонент и *i*-ая компонента интерпретируется как степень наличия тематики в документе.

Одним из главных недостатков латентно-семантического индексирования является несоответствие вероятностной модели реальности. Задача построения вероятностно-тематической модели коллекции сводится к оценке параметров распределения

$$p(w,t) = \sum_{t \in T} p(t)\, p(w|t)\, p(d|t), \tag{8}$$

где $p(t)$ – неизвестное априорное распределение тем во всей коллекции, $p(d)$ – априорное распределение на множестве документов, а $p(w)$ – априорное распределение на множестве слов.

В 2003 г. была предложена модель Latent Dirichlet Allocation [4], в которую было добавлено предположение о том, что распределение тематик по документам $p(t|d)$ и распределение слов по тематикам $p(w|t)$ порождены распределениями $Dir(\theta, \alpha)$ и $Dir(\theta, \beta)$.

Модель характеризуется количеством тематик T и параметрами распределений α и β.

Модели дистрибутивной семантики. В 2010 г. Томашом Миколовым была предложена модель дистрибутивной семантики word2vec [5], основывающаяся на гипотезе, что контекст слова определяется его окружением. Существуют две разновидности модели word2vec – Continuous Bag of Words и SkipGram. Первая решает задачу предсказания слова w_i на основании контекста (ближайших слов), вторая модель является противоположной – зная слово w_i, найти его контекст. Данная задача ставится как максимизация средней лог-вероятности нахождения слова в контексте:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{0<|j|\le c} \log p(w_{t+j} \mid w_t). \tag{9}$$

Наиболее распространенным способом получения вектора документа является суммирование или усреднение векторов слов, из которых состоит документ [6, 7].

Позже Томаш Миколов развил идею word2vec и предложил модель paragraph2vec [8], в которую помимо векторов слов входят векторы документа. Были предложены две модели – Distributed Memory и Distributed Bag of Words. Модель DM прогнозирует слово по известным предшествующим словам и вектору документа, DBOW прогнозирует случайные группы слов в абзаце только на основании вектора документа. Следует отметить также и отечественные работы по применению word2vec, в том числе работу [9] по повышению качества рекомендаций за счёт генерации искусственных негативных примеров в коллаборативной фильтрации.

Результаты исследования и их обсуждение

Предлагаемый подход состоит в построении word2vec-модели Distributed Bag-of-Words и усреднении векторов слов из описаний для построения признакового описания рассматриваемого объекта. В данной работе подход тестируется в задаче рекомендаций фильмов.

Для проведения экспериментов был выбран набор данных MovieLens 1M [10]. Данный набор содержит данные о 1 000 000 оценок 6040 пользователей 3952 фильмам. Матрица оценок содержит 4,19% ненулевых элементов. Для каждого фильма через IMDB API была получена его сюжетная аннотация. Помимо этого, для увеличения размера корпуса и, как следствие, улучшения векторных представлений текстов, были взяты дополнительные сюжетные аннотации 500 тыс. других фильмов.

Все тексты были предварительно обработаны, приведены к нижнему регистру, кроме того были отброшены стоп-слова, которые встречались более чем в 50% документов, а также выброшены слова, встречавшиеся менее чем в 1% документов.

Для обучения модели word2vec была выбрана размерность 300, длина контекста – 8.

Ниже представлена таблица, в которой для некоторых слов подобраны семантически близкие слова. Полученная таблица свидетельствует о высоком качестве векторных представлений слов.

Для построения вероятностно-тематической модели LDA было выбрано 50 тематик.

В табл. 2 представлены ключевые слова для некоторых тематик, построенной LDA-модели.

TF-IDF модель была построена с использованием биграмм и триграмм.

В качестве признаковых описаний объектов использовались векторы, полученные различными моделями представления текстов на естественном языке:

- TF-IDF;
- LSI с размерностью 50 тематик;
- LDA с размерностью 50 тематик;
- word2vec с размерностью вектора 300 и усреднением векторов слов;

Оценка проводилась при помощи перекрёстной проверки на 5 блоках, для оценки близости векторов использовалось косинусное расстояние. Измерение качества полученных моделей проводилось с использованием метрик *RMSE*, F_1@5 и F_1@10.

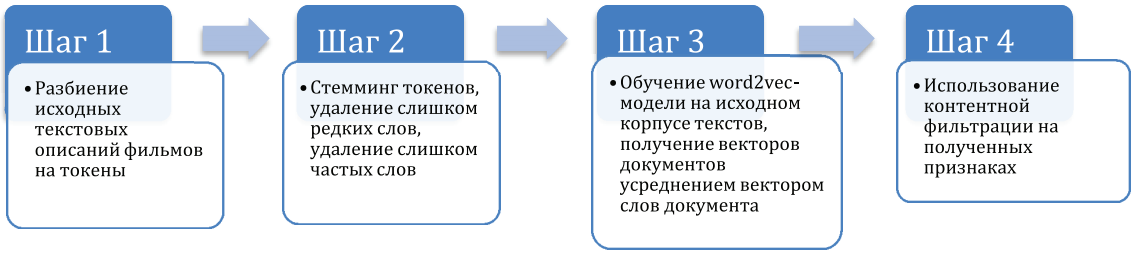

Иллюстрация шагов предложенного алгоритма

Таблица 1

Семантически близкие слова, полученные моделью word2vec

Titanic	Расстояние	Terminator	Расстояние	Godfather	Расстояние	Casino	Расстояние
RMS	0,790201	Half-Life	0,691311	Exorcist	0,674286	roadhouse	0,727415
Lusitania	0,687613	Aquaman	0,682283	Bastard	0,670992	nightclub	0,724142
Hindenburg	0,670862	Mutant	0,669808	Corleone	0,663091	hotel	0,70251
HMS	0,670684	Filter	0,666307	Sopranos	0,654051	bank	0,698563
liner	0,662014	Predator	0,663646	Hobo	0,653354	racetrack	0,678729
USS	0,658244	X-Men	0,661398	Rockers	0,633333	marina	0,678018
Vasa	0,640993	Puppets	0,658903	Ghoulash	0,62528	club	0,674167
S.S.	0,636868	Blade	0,656534	Notorious	0,620483	bar	0,669398
Flight	0,631798	Returns	0,649444	Imposter	0,619021	saloon	0,662977
submarine	0,630688	PS2	0,645625	Disciples	0,614807	casinos	0,651787

Таблица 2

Ключевые слова для некоторых тематик, полученных LDA-моделью

№	Ключевые слова	Вероятная тематика
1	film, movi, stori, play, show, perform, charact, scene, music, star, follow, includ, peopl, around, end, song, first, band, act, role	Театр/драма
2	polic, kill, murder, offic, prison, arrestm investig, man, gang, case, death, crime, one, drug, shoot, killer, crimin, detect, escap, suspect	Криминал/триллер
3	war, american, forc, state, soldier, armi, unit, order, offic, german, govern, british, world, command, general, men, captain, group, agent, militari	Военные фильмы
4	tell, go, ask, see, say, leav, back, day, get, call, goe, come, home, next, want, find, night, one, know, time	Не поддаётся интерпретации
5	use, ship, destroy, island, human, earth, crew, attack, discov, world, control, escap, power, one, alien, monster, caus, find, time, rescu	Приключенческие фильмы
6	king, power, kill, find, use, one, evil, return, take, villag, save, princ, magic, howev, queen, order, fight, name, world, death	Исторические драмы

Таблица 3

Результаты перекрёстной проверки качества модели контентных рекомендаций

	RMSE	F_1@5	F_1@10
TF-IDF	0,9587	0,4509	0,5298
TF-IDF с n-граммами	0,9242	0,4684	0,5361
LSI	0,8913	0,4645	0,5393
LDA	0,9102	0,4715	0,5469
word2vec	**0,8798**	**0,5056**	**0,5757**

Представлена табл. 3 с результатами оценки качества модели на перекрёстной проверке по пяти блокам для рассматриваемых метрик.

Как видно из табл. 3, предложенный подход показывал высокое качество рекомендаций. Однако, так как данный подход был опробован только на описаниях фильмов, требуется провести дополнительные исследования данного подхода на других тематиках.

Заключение

В данной работе были рассмотрены способы построения контентных рекомендаций фильмов на основе различных методов построения векторных представлений текстов на естественном языке, а также предложена модель построения контентных рекомендаций на основе модели word2vec. Предложенный подход показал высокое качество в задаче рекомендаций фильмов на наборе данных MovieLens 1M.

Список литературы

1. Pazzani M., Billsus D. Content-based recommendation systems // The adaptive web. – 2007. – С. 325–341.

2. Said A. et al. Recommender systems evaluation // Workshop on Recommendation Utility Evaluation: Beyond RMSE, RUE 2012-Workshop at the 6th ACM International Conference on Recommender Systems, RecSys 2012. – 2012. – С. 21–23.

3. Palmer D.D. Text preprocessing // Handbook of Natural Language Processing, Second Edition. – Chapman and Hall/CRC, 2010. – С. 9–30.

4. Blei D.M., Ng A.Y., Jordan M.I. Latent dirichlet allocation // Journal of machine Learning research. – 2003. – Т. 3, № Jan. – С. 993–1022.

5. Mikolov T. et al. Distributed representations of words and phrases and their compositionality // Advances in neural information processing systems. – 2013. – С. 3111–3119.

6. Musto C. et al. Word Embedding Techniques for Content-based Recommender Systems: An Empirical Evaluation // RecSys Posters. – 2015. URL: http://ceur-ws.org/Vol-1441/recsys2015_poster23.pdf (дата обращения: 15.01.2018).

7. Kusner M. et al. From word embeddings to document distances // International Conference on Machine Learning. – 2015. – С. 957–966.

8. Le Q.V., Mikolov T. Distributed Representations of Sentences and Documents // ICML. – 2014. – Т. 14. – С. 1188–1196.

9. Беляков Д.Е., Кантор В.В. Исследование эффекта добавления негативного сэмплирования при обучении факторизационных машин в задачах построения рекомендательных систем // Информационные процессы. – 2017. – Т. 17, № 2. – С. 159–163.

10. Harper F.M., Konstan J.A. The movielens datasets: History and context // ACM Transactions on Interactive Intelligent Systems (TiiS). – 2016. – Т. 5, № 4. – С. 19.